

Amendments to the Specification

On page 2, line 3, please replace the paragraph and substitute the following:

"One potential problem with using a ~~simulator~~ an emulator that can interact with a host computer is contamination, whether the contamination be from a virus or just involves changing the operating parameters of the host or the ~~simulator~~ emulator. If such a ~~simulator~~ an emulator were provided, it would need to insulate the host computer from the ~~simulator~~ emulator and the ~~simulator~~ emulator from the host computer, while providing full functionality of both."

On page 2, line 28 through page 4, line 21, please replace the text with the following:

"An operating environment ~~simulation~~ emulation system is shown in Figure 1. This ~~simulation~~ emulation system will be referred to as such, an emulator ~~simulator~~ or a memory device. One of its functions is as a transportable volume acting as a local environment repository. The operating environments it may emulate would be those commonly used by computer users, including Windows™ 95/98/2000/NT, referred to here as Windows™ XX, Unix, Linux and MacIntosh. Those operating systems that run on the IBM-PC compatible standard will be referred to as personal computer (PC) compatible. The system 10 includes a memory 12 and a connector 14.

The memory 12 includes a set of instructions, referred to here as an emulator ~~simulator~~ 18. There may be several versions of the ~~simulator~~ emulator installed in the memory device 12. Each copy of the ~~simulator~~ emulator is intended for a different operating system and/or processor combination on the host computer. The combinations of processor and operating system are only limited by the manner in which the ~~simulators~~ emulators are packaged. For example, there could be a ~~simulator~~ emulator packaged by processor, such as the Intel Pentium 3™. The memory device may have several copies of the ~~simulator~~ emulator on it, one each for Windows XX, Linux and Unix. Alternatively, the ~~simulators~~ emulators could be packaged by operating system, such as one for Windows XX, with several versions for common processors.

Within the memory device is a data file 16. The data file 16 contains all of the elements of the original operating environment to be emulated ~~simulated~~ by the host computer. The data file will typically remain the same, as it is accessible by the ~~simulators~~ emulators. When connected to the host computer 20, the operating system of the host computer runs the emulator ~~simulator~~ as a task and accesses the data file. Once the data file is accessed, the host computer becomes an emulation ~~simulation~~ of the original operating

environment accessing the data file in the manner of the particular operating system selected, as defined by the executable file.

It must be noted that the host computer does not have to be a complete personal computer system. The ~~simulator~~ emulator only requires a processor 22 that is running an operating system and a connector. This could include any number of accessories available today, such as palmtop devices, notebook computers, possibly even cellular phones, if the phone has the correct connectors, operating system and processor. These non-PC devices will be referred to as accessory devices.

Similarly, the cable 24 may not be necessary depending upon the standard used to establish communication between the two systems. If the standard used is IEEE (Institute of Electrical and Electronic Engineers) 1394, also known as 'firewire,' or USB (Universal Serial Bus), a cable will be needed. Other types of connections, such as wireless communications in accordance with a given wireless standard such as 802.11b, Ethernet, or infrared connections could also be made. The connectors in those examples would be of the appropriate technology to allow information to pass between the host computer and the ~~simulation~~ emulation system. In some embodiments, a USB cable that can connect directly to the host computer may be the easiest and most convenient configuration. The user can connect the ~~simulation~~ system by simply plugging in the cable to the host computer.

In the below discussion of the operation of the invention, there will be two different operating systems or operating system versions discussed. The host computer to which the ~~simulation~~ emulation system connects and executes the ~~emulator~~ simulator will have an original operating system. This is the operating system under which the host computer is running when the ~~simulator~~ emulator is loaded onto the host computer. The ~~simulated~~ emulated operating system is that operating system being ~~simulated~~ emulated by the ~~simulation~~ emulation system. The two operating systems may be of different types and/or versions, as will be discussed below.

Figure 2 shows one embodiment of a method of establishing a ~~simulated~~ emulated operating environment upon connection between the memory device and the host computer. At 26, the connection is made. Using a plug-and-play sequence familiar to most users, the host computer will display the new device as an attached memory device, with a list of executable files. The list of executable files is the various versions of the ~~simulators~~ emulators available on the memory device in the ~~simulation~~ emulation system. For example, the user may connect the ~~simulation~~ emulation system as drive 'E:' and display a list of files

on that drive. The files may be named so as to make the identification of the appropriate file easy on the user, such as 'Windows XX.exe' or 'Linux.exe.'

The host system receives the user input and loads the version of the ~~simulator~~ emulator desired, from the memory of the ~~simulation~~ emulation system to the host processor at 30. The host processor then executes the selected set of instructions at 32. Upon execution of the selected ~~emulator~~ simulator, the ~~simulated~~ emulated operating environment is established on the host computer or accessory device.

The system allows the user to exactly replicate the typical computing environment in which that user operates. The user may use Windows NT™ at work. Desiring to work at home in the same operating system, the user takes his ~~simulator~~ emulator and plugs it into his home computer. The home computer may have some other member of his family running a task, such as a download in Windows 98™. A pop-up window appears inquiring as to the selection of an executable file from the new device. The user selects the proper ~~emulator~~ simulator. The selection and execution of the proper ~~simulator~~ emulator will cause the host computer to appear to be a Windows NT™ system, even though the original operating system of the host computer is Windows 98™."

Please replace the text starting on page 5, line 3, through page 6, line 14 with the following:

"However selected, the ~~simulator~~ emulator then provides the processor with the necessary data and applications to run as if it were a Windows NT™ system. The user can now move around and perform tasks in the same environment as he uses at work. No preconfiguration or modification of the host computer is necessary to run the ~~simulated~~ emulated operating system.

One concern with this approach is any possible contamination or interference between the ~~simulator~~ emulator and the host computer. Figure 3 shows a flowchart of one embodiment of a method to prevent this kind of contamination. The set of instructions that comprise the ~~simulator~~ emulator will also operate to insulate the two systems to prevent any inadvertent or intentional interaction between them.

As can be seen in Figure 3, any type of task management available to the primary operating system will be disabled at 34. For example, the primary operating system may be Windows 98™, as in the above example. In this operating system, it is possible to press *ALT-TAB* to switch between tasks running in the primary operating system. Other types of task management are available, depending upon the operating system. Conceivably, one could press *ALT-TAB* while the secondary operating system is running from the ~~simulator~~

emulator and switch to other tasks running on the primary operating system. However, with the ~~simulator~~ emulator software in place, this type of interaction may be prevented.

It is possible that the user may want some sort of interaction between the two systems. For example, the user may have worked on a file at the office that he wants to save onto his home computer. The ~~simulator~~ emulator may allow some sort of user selection to define the nature of allowed interactions. In order to access the underlying system, various types of security could be required, such as passwords. However, for this example, complete insulation will be assumed.

In order to ensure complete insulation, input devices will have to have their interrupts routed to run only through the secondary operating system at 36. Devices such as keyboards, joysticks and mice generate an interrupt or other type of signal that notifies the host processor of an input signal. Once the ~~simulator~~ emulator starts to operate, all input/output signals will be routed through it to allow the ~~simulator~~ emulator to enforce the insulation. This will prevent any contamination and will allow tasks running on the primary operating system to remain undisturbed.

As a failsafe to ensure insulation, an environmental shut down can be provided that allows the host system to protect itself at 38. Inadvertent or intentional actions could cause the insulation between the systems to be breached. If that happens, the ~~simulator~~ emulator will effect an environmental shutdown of the secondary operating system as shown at 38. Otherwise the system continues to operate.

In this manner, users are provided with the ability to preserve a current operating system environment and transport it to another computer or computing device. The ~~simulator~~ emulator can be connected to any computing device with a processor and the proper connector. Users can now transport their local environment with a smaller, lighter, more convenient device than a typical laptop computer.

Thus, although there has been described to this point a particular embodiment for a method and apparatus for an operating environment ~~simulation~~ emulation system, it is not intended that such specific references be considered as limitations upon the scope of this invention except in-so-far as set forth in the following claims."